

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 11-134155

(43)Date of publication of application : 21.05.1999

(51)Int.Cl.

G06F 3/14
G06F 17/30

(21)Application number : 09-297214

(71)Applicant : MITSUBISHI ELECTRIC CORP

(22)Date of filing : 29.10.1997

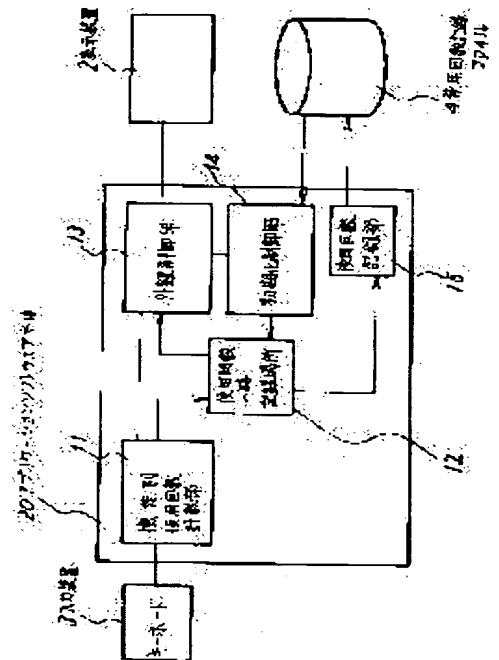
(72)Inventor : SAWADA TORU

(54) SOFTWARE CONSTITUTING METHOD WITH CHANGING FUNCTIONS ACCORDING TO USE FREQUENCY

(57)Abstract:

PROBLEM TO BE SOLVED: To change contents of function and also operation screens by additionally displaying a new screen each time a frequency of use for each function exceeds a predetermined specific frequency.

SOLUTION: When application software is activated, an initialization control part 14 reads in the frequency of use from a use-frequency recording file 4, records it in a use-frequency temporary recording location 12, and then calls an outward appearance control part 13 to display prepared outward appearances according to the use-frequency recorded in the use-frequency temporary recording location 12. Each time a user selects a function in a top menu or selecting a function by pressing a button, a use-frequency-by function counting part 11 continues to add the use-frequency in the use-frequency temporary storage location 12 and also displays to the user a message indicating that the outward appearance of the activating screen of the application software is changed when the total value of use-frequencies exceeds a threshold, and a function outward appearance control part 13 is called to change the outward appearance.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平11-134155

(43) 公開日 平成11年(1999) 5月21日

(51) Int.Cl.⁶
G 0 6 F 3/14
17/30

識別記号
3 3 0

F I
G 0 6 F 3/14 3 3 0 A
15/403 3 4 0 B

審査請求 未請求 請求項の数 3 O L (全 16 頁)

(21) 出願番号 特願平9-297214

(22) 出願日 平成9年(1997)10月29日

(71) 出願人 000006013

三菱電機株式会社

東京都千代田区丸の内二丁目2番3号

(72) 発明者 澤田 徹

兵庫県神戸市兵庫区浜山通6丁目1番2号

三菱電機コントロールソフトウェア株式
会社内

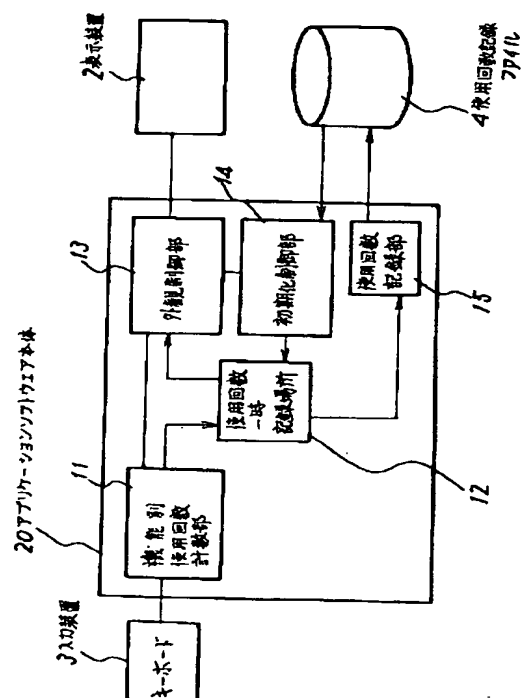
(74) 代理人 弁理士 宮田 金雄 (外2名)

(54) 【発明の名称】 使用頻度に応じて機能が変化するソフトウェアの構成方法

(57) 【要約】

【課題】 ソフトウェアを操作するための操作画面において、表示される機能選択ボタンの数は、オペレータがこのソフトウェアの操作に習熟しているか否かに関わりなく、常に一定であった。そのため、操作になれていない初心者にも、ベテランしか使用しない使用頻度の少ない機能選択ボタンまでが表示され、操作画面が複雑となって操作を難しくしていた。そこで、オペレータの習熟度に応じて、表示される操作画面の構成が変化し、オペレータの技量にあった表示画面が得られるようにしたい。

【解決手段】 表示装置2の画面に表示される機能選択ボタン毎に機能別使用回数計数部11により使用回数を累積計数して、使用回数記録ファイル4に記録する。使用回数があらかじめ定めた閾値を越えることに、外觀制御部13があらかじめ定めた追加機能の選択ボタンを次々に表示するようにする。



【特許請求の範囲】

【請求項1】 ソフトウェアを操作するための画面を表示するプログラムを有するコンピュータ用ソフトウェアプログラムの前記画面に表示される個々の機能ごとに、機能別使用回数を計数し記憶する使用回数計数手順と、上記機能別使用回数があらかじめ定めた所定の回数を超えるごとに、該機能ごとに、該ソフトウェアプログラム内にあらかじめ用意され、且つ、前記画面には表示されていなかった画面を前記画面内に追加表示する画面追加手順とを有し、使用頻度に応じて機能が変化することを特徴とするソフトウェアの構成方法。

【請求項2】 利用者を特定するための利用者識別手順を有し、使用回数計数手順は前記利用者識別手順によって識別した利用者別に機能別使用回数を計数、記憶し、画面追加手順は前記利用者別に画面内に表示する追加画面を決定することを特徴とする請求項1に記載の使用頻度に応じて機能が変化するソフトウェアの構成方法。

【請求項3】 機能別使用回数の計数は、該ソフトウェアが使用されている時間中に該機能が使用されない状態があらかじめ定めた所定の時間長さに達すること、該機能の使用回数を減算計数する減算計数手順を含むものであり、

画面追加手順は前記機能別使用回数があらかじめ定めた所定の回数を下回ること、追加していた画面を削除する画面削除手順を含むものであることを特徴とする請求項1に記載の使用頻度に応じて機能が変化するソフトウェアの構成方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】この発明はコンピュータにおけるソフトウェア例えばアプリケーションソフトウェアの機能の構成方法の改良に関するものである。

【0002】

【従来の技術】図18は例えば特開平6-180638号公報に示された、従来のソフトウェアの操作画面を変化させる装置のシステム構成図である。図に於いて、1はシステム制御部であり、その中にモニタ或はディスプレイと呼ぶ表示装置2を制御するための表示制御部91、キーボード3からの入力を制御する入力制御部92、以下に説明するソフトウェアの使用回数をカウントして必要な制御を行う使用回数処理部93、優先度算出処理部94及びユーザ設定処理部95を備え、それぞれはハードウェアとソフトウェアによって構成されており、またこのシステム制御部1にはCRTやLCD等からなる表示装置2と、操作制御を行うためのキーボード3（マウスなども含む）と、フロッピーディスクやハードディスク、光ディスク或は半導体メモリ等からなる記憶装置4が接続されている。

【0003】表示装置2は表示制御部91によって制御され、機能を表わすボタンや機能名等、例えば図19の

30に示す様な機能を表わすボタンの図が多数表示されている。これらのボタン30は機能を指定する時、マウス等のポインティングデバイスで選択するか、或は表示装置にタッチパネル機能を付加しておき、指で触れることによって選択する。

【0004】図20は上記システムを制御する為のソフトウェアの主要部分のフローチャートである。以下、この図に従って処理手順を説明する。スタートすると、各機能選択（ボタン操作）毎に、各機能別に何回使用されたかわかるようにしておくために、使用回数処理部93によって使用回数が計数され、メモリ上に一旦記憶し（#1）、次に外部記憶装置4内にある図18に示す管理テーブル41（図21に詳細を示す）に登録する（#2）。

【0005】上記メモリ上の管理テーブルも使用回数処理部93内にある。図21に示す上記管理テーブルの各エントリは全ての機能選択ボタン30（以下ボタンという）に対応して設けられており、例えば、ボタン0は使用回数5回、ボタン1は使用回数7回、ボタン2は使用回数12回……というように登録される。

【0006】続いて、優先度算出処理部94において、上記登録された内容に基づき各ボタン30について使用回数の多い順に並び換えが行なわれる（#3）。図22は使用回数の多い順にボタン30（機能名等）を並び換えたテーブルの例であり、このテーブルは優先度算出処理部94内にある。こうして、使用回数が最も多いボタン（機能）を選出し（#5）、画面上の適当な場所に、そのボタンを例えば使用回数の多い順に表示する（#10）。

【0007】上記のように選出されたボタンは、以後、少なくともシステムの状態が機能指定可能な状態にあるときは常に表示され、その中から特定のボタン（機能）が指定されるとシステム内のソフトウェア群（図示していない）から、この選択されたボタン（機能）に対応したソフトウェアが呼び出され、ただちに実行される。使用回数が多い順にボタンが表示されると、操作の多くは最も上に配置されたボタンを選択することとなり、操作がし易くなるわけである。

【0008】

【発明が解決しようとする課題】従来の操作画面の変更方法は以上のように構成され、仕組みまれているので、使用回数の多いボタン（機能）の選択を容易にするだけであった。このため、使用回数が多い機能であっても、機能の内容（表示画面の形）は変わらず、慣れないあいだだけ必要な画面の表示が、操作になれて不要になってからも表示され、かえって煩わしいなど、本質的に使い易くなるものではないという問題点があった。

【0009】この発明は上記のような課題を解決するためになされたものであり、使用回数が高い機能をアプリケーションソフトウェアが自身で認知し、機能内容（或

は表示画面の形)を、例えば使用回数(習熟度)に応じて変化することで、良好な操作画面の変更方法を得ることを目的とする。

【0010】

【課題を解決するための手段】この発明の使用頻度に応じて機能が変化するソフトウェアの構成方法は、操作するための画面を表示するプログラムを有するコンピュータ用ソフトウェアプログラムの、前記画面に表示される個々の機能ごとに、機能別使用回数を計数し記憶する使用回数計数手順と、上記機能別使用回数があらかじめ定めた所定の回数を超えるごとに、該機能ごとに、該ソフトウェアプログラム内にあらかじめ用意され、且つ、前記画面には表示されていなかった画面を前記画面内に追加表示する画面追加手順とを有するものである。画面追加手段は、今まで表示されていなかった新たな画面(新たな機能)を追加表示するので、ソフトウェアの操作機能そのものを新しくする作用がある。

【0011】また、この発明の使用頻度に応じて機能が変化するソフトウェアの構成方法は、利用者を特定するための利用者識別手順を有し、使用回数計数手順は前記利用者識別手順によって識別した利用者別に機能別使用回数を計数、記憶し、画面追加手順は前記利用者別に画面内に表示する追加画面を決定するものである。利用者識別手順は、機能別使用回数の計数および画面追加手順をそれぞれ利用者別に対応するようにさせるので、変化する機能が利用者別に異なった変化をするように作用する。

【0012】また、この発明の使用頻度に応じて機能が変化するソフトウェアの構成方法は、機能別使用回数の計数が該ソフトウェアが使用されている時間中に該機能が使用されない状態があらかじめ定めた所定の時間長さに達すると、該機能の使用回数を、所定の回数減算計数する減算計数手順を含むものであり、画面追加手順は前記機能別使用回数があらかじめ定めた所定の回数を下回ると、追加していた画面を削除する画面削除手順を含むものである。減算計数手順は、使用しない機能に対応する追加済の機能を削除し、画面を簡素化するように作用する。

【0013】

【発明の実施の形態】

実施の形態1. 以下、この発明の実施の形態1を図に基づいて説明する。図1はこの発明のシステムの機能ブロック図である。図1において、20はアプリケーションソフトウェア本体を示し、その中に以下の要素が含まれている。

【0014】即ち、11はアプリケーションソフトウェアの個々の機能ごと(機能毎の意味は押しボタン1つずつの意である)に使用回数を計数する機能別使用回数計数部、12はその計数した使用回数を一時的に記録する使用回数一時記録場所、13はその使用回数に応じてア

プリケーションソフトウェアの外観を変化させる外観制御部、14はアプリケーションソフトウェア起動時に記憶装置4から使用回数を読み込み外観制御部13を呼び出す初期化制御部、15は使用回数一時記録場所の使用回数を記憶装置4に記録する使用回数記録部である。2は画面表示装置(モニタ)、3はキーボードやマウス等の入力装置、4は記憶装置(ハードディスク等)を示しているが、以下の説明では使用回数記録ファイルという。

10 【0015】次に動作について説明する。図2は上記システムを制御するためのソフトウェアの主要部分のフローチャートである。以下、この図に従って処理手順を説明する。まずアプリケーションソフトウェアを起動すると、初期化制御部14は、使用回数記録ファイル4から使用回数を読み込み(S1)、使用回数一時記録場所12に記録し(S2)、(または使用回数記録ファイル4が無い場合はファイル4を新規作成して使用回数一時記録場所12に使用回数を0回と記録し)、続いて外観制御部13を呼び出す。外観制御部13は使用回数一時記録場所12に記録されている使用回数に応じてあらかじめ用意してある外観を表示する(S3)。

20 【0016】利用者がアプリケーションソフトウェアの各機能をメニューから選択するか、またはボタンを押すなどして機能を選択する毎に(S5)、機能別使用回数計数部11は使用回数一時記録場所12の使用回数を加算し続けると同時に(S6)、使用回数の合計値が閾値を超えた場合は(S9)アプリケーションソフトウェアの起動用画面の外観を変化させる旨を利用者にメッセージとして表示し(S10)、機能外観制御部13を呼び出して外観を変化させる。

30 【0017】図3の50は上記(S10)に説明した利用者に表示されるメッセージの例、図4の51は外観を変化させる前の(例えば最初にS3で表示する)メッセージ画面、図5の52は外観を変化させた後の(例えば何回目かのS3で表示する)メッセージ画面の例であり、使用回数に応じて図4から図5へ変化する様子を示す。図5の52の下半分は図4の51では表示されていなかったものである。アプリケーションソフトウェアの使用を利用者が終了すると(S5)、使用回数記録部15は使用回数一時記録場所12の使用回数の記録を使用回数記録ファイル4に書き込む(S7)。

40 【0018】ステップS4、S5、S6、S7はこの発明に言う使用回数計数手順である。ステップS9、S10、およびS3はこの発明に言う画面追加手順である。

50 【0019】実施の形態2. 実施の形態1における。フローチャート図2のステップS3について、アプリケーションソフトウェアの個々の機能ごとに操作画面の外観を変化させる方法の具体的説明を図6～図8の画面例と図9のフローチャートにより述べる。図7の画面31(図6の31と同じ)は、例えば従来例の図17のボタ

ンを選択することにより表示されるもので、機能毎の表示画面という。機能ごとの表示画面31の外観を変化させるには、機能ごとにウインドウを用意して、そのウインドウ(図6の30)の上位部分(図6の31)に基本的な選択肢・ボタンなどを配置し、下位部分(図6の32、33)に応用的な選択肢・ボタンなどを配置する。アプリケーションソフトウェア利用時の最初の頃(即ち使用回数が少ないとき)は図7の画面31に示すようにウインドウの上位部分31だけを表示しておいて、使用回数が閾値を超える毎にウインドウの表示する面積を下位方向に増やして行くことで実現する。この方法であれば、ウインドウを1枚用意するだけで使用回数による変化を簡単に実現することができる。なお図中99は説明のため画面を区分する説明補助線である。

【0020】図9は図2の処理のフローチャートのS9～S10～S3を詳細に示す図、図6、図7はアプリケーションソフトウェアの個々の機能の外観の変化の例、図8の81は利用者に表示されるメッセージの例である。ここでは、検索機能のウインドウが変化する様子を示している。図9のフローチャートについて説明する。ステップS9、S10は、図2のステップS9とS10そのものを示している。ステップS9でこの機能の使用回数があらかじめ決めてある所定の閾値を越えたことが判明すれば図8の81に示すような変化メッセージを表示する(S10)。次にステップS13で、図6の画面の表示部分31部分を32部分まで広げ、ステップS14で表示する。

【0021】このようにすることの効果は、使用回数に応じて機能を変化させる処理において、あらかじめ用意しておくウインドウの外観は1通りだけでよいということと、機能の変化を示すための表示の変更には、ウインドウの表示する部分のサイズを変えるだけでよいので、実現が容易であることである。

【0022】実施の形態3. 図2のフローのステップS9～S10～S3を具体化する別の方法として、機能に対して複数のウインドウ画面(例えば図10、11、12、13に順に変化画面の例を示す)を用意して、機能を実行するためには利用者がそのウインドウを順番に見ていき必要な入力を行なう形態とし、順番の最初に近いウインドウ(図10の33)には基本的な選択肢・ボタンなどを配置し、順番が末尾寄りのウインドウには応用的な選択肢・ボタンなどを配置し、そして末尾のウインドウ(図13の36)に機能を実行するボタンを配置し、アプリケーションソフトウェア利用時の最初の頃は最初と末尾のウインドウ2枚だけ表示しておいて、使用回数が閾値を超える毎に中間のウインドウの表示をするように変化させていく(ウインドウの表示枚数を増やしていく)ことで実現することもできる。

【0023】この方式であれば、ウインドウの表示枚数を変えるだけで使用回数による変化を簡単に実現するこ

とができる。図14は処理のフローチャート、図10、図11、図12、図13は機能に対して用意する複数枚のウインドウの例を示している。

【0024】図14のフローについて説明する。図中ステップS9は図2のステップS9と同じものを示している(S10は記載を省略)。ステップS9で機能を実行した回数が、あらかじめ定めた所定の閾値を越えたと判断されると、変数Nを1と設定する(ステップS22)。ステップS23でNが、あらかじめ用意した表示画面の数以下であれば、ステップS25でN枚目の画面を表示する。

【0025】又、ステップS23でNが表示画面の最大数と同等以上であればステップS24で用意している一連の画面の最後の表示画面を表示し、オペレータはステップS26でその機能を実行する。この方法の効果は、一連の流れをもつ機能について、その表示画面の外観を変化させる処理を簡単に実現できることにある。

【0026】実施の形態4. 本発明の実施の形態4による使用頻度に応じて機能が変化するソフトウェアの構成方法をとるシステムの構成図を図15に示す。図中40は利用者識別部である。利用者識別部40はいわゆるOSシステムコールを利用するなどしてアプリケーションソフトウェアの利用者名を得る働きをするものである。第1に利用者ごとに使用回数記録ファイル41を用意することで、第2にアプリケーションソフトウェアの起動時に初期化制御部14が利用者識別部40を呼び出しアプリケーションソフトウェアの利用者名を得たのち、その利用者用の使用回数記録ファイル41から使用回数を読み込んで使用回数一時記録場所12に記録するようにすることで、第3にアプリケーションソフトウェア終了時に使用回数記録部15が使用回数一時記録場所12の使用回数を利用者用の使用回数記録ファイル41に書き込むようにすることで、アプリケーションソフトウェアの利用者一人一人に対応する使用回数を計数および記録することができ、利用者ごとに個別に操作画面表示の外観の変化を見ることが可能となる。機能別使用回数を利用者別に計数する手順を利用者識別手順という。

【0027】このようにすることの効果は、アプリケーションソフトウェアが利用者一人一人の使用回数に応じて別個の変化をすることができることにより、利用者には「私が育てたアプリケーションソフトウェアである」、「私だけのアプリケーションソフトウェアである」という印象を持たせて、アプリケーションソフトウェアに対する愛着感を持たせることができることである。

【0028】実施の形態5. この発明の実施の形態5について図16、17により説明する。図16は構成図であり、図において22は機能別不使用時間計数部である。これ以外の部分は図1と同じであるので詳細な説明を省略する。機能別不使用時間計数部22は、このシステムに内蔵される図示しない時計装置からの信号によっ

て、このソフトウェアが立ち上げられているのに、その機能が一度も使用されていない時間を各機能別に計数する。そしていずれかの機能についてこの時間が所定の値（例えば10時間）を越えるごとに、機能別計数回路11に信号を送って、その機能別使用回数を1回減算する。ここで不使用時間とは、このソフトウェアそのものが使用されていない時間は含まないものである。

【0029】図17に図16の機能別不使用時間計数部22の動作フローチャートを示す。図17のステップS30とステップS31とステップS32が、機能別不使用時間計数部22の動作を示すもので、その他のステップは図2のフローと同じなので詳細な説明は省略する。ステップ3で過去の使用回数に合わせた操作画面を表示している。この状態でこの画面に表示している各機能について不使用時間を計数する（S30）。この計数時間が所定の値（例えば10時間）になるごとに、減算信号を1回出力する（S31）。所定の時間に達する前に当機能が使用されると計数していた時間はリセットされ再度ゼロからカウントが開始される。上記出力された減算信号は過去の累積使用回数を1回減らすことになる（S32）。

【0030】仮に、今、過去にしばしば使用された結果、使用回数が閾値を越え特定の画面（例えば熟練者用の機能を含む画面）が表示されるようになっていた機能が、全く使用されなくなった場合、この状態が長く続くと、累積使用回数がやがて減少し所定の閾値を下回ると、表示される画面は再びもとの、例えば非熟練者用の簡単な機能しか含まない画面にもどされる、あるいは、追加されていた画面は削除されるのである。これによって、使用者の習熟度の進退に応じて最適な画面が表示されるようになる。ステップS30～S32は本発明に言う減算計数手順である。追加されていた画面が削除される手順（画面追加と実質的には同じである）は本発明に言う画面削除手順である。

【0031】以上、実施の形態1～5の説明では、アプリケーションソフトウェアが1つとして説明したが、複数個あっても同様である。また、使用回数の加算計数、減算計数はそれぞれ一定値を越えれば意味がないので、一定値以上はカウントしないようにしても良い。説明における閾値は1つとは限らず、機能の複雑さに応じて複数レベルあって良いことは言うまでもない。

【0032】

【発明の効果】この発明の使用頻度に応じて機能が変化するソフトウェアの構成方法によれば、個々の機能毎にその使用回数が所定の値を越えると、表示画面を変化させるようにしたので使用頻度に応じてソフトウェアの使用可能な機能が充実化されるという効果がある。

【0033】又、変化する画面は、多くの機能の操作表示を含んだ1つの画面の表示面積を変化させることで得られるので、画面用プログラムの作成が容易である。

又、あらかじめ用意した複数の画面を次々と切替えて使用するので、画面の作成、追加、修正等の作業が容易である。

【0034】又、変化する画面は、ソフトウェアを利用する利用者別に使用回数を計数し、利用者別に表示画面を決定するので、利用者の用途に応じた最適の操作画面が表示できると言う効果がある。

【0035】また、ソフトウェアの使用者が特定の機能を長期に渡って使用しなければ、追加していた機能（ボタン画面）を削除して元に戻すようにしたので、その機能を使用する習熟度の上昇下降に応じて、新たな機能を追加したり削除したりすることができ、習熟度に応じた最適な機能を提供することができる。

【図面の簡単な説明】

【図1】 この発明の実施の形態1のソフトウェア操作画面の変更システムを示す機能ブロック図。

【図2】 図1のものの動作フローチャートである。

【図3】 図1のものに於て、表示されるメッセージ画面である。

【図4】 図1のものに於て、表示される操作表示画面である。

【図5】 図1のものに於て使用回数が多くなったとき表示される操作表示画面である。

【図6】 実施の形態2による機能表示画面である。

【図7】 図6の画面の表示形態を示す図である。

【図8】 実施の形態2によるメッセージ画面である。

【図9】 実施の形態2による動作のフローチャートである。

【図10】 実施の形態3による1枚目の表示画面である。

【図11】 実施の形態3による2枚目の表示画面である。

【図12】 実施の形態3による3枚目の表示画面である。

【図13】 実施の形態3による4枚目の表示画面である。

【図14】 実施の形態3の処理手順を示すフローチャートである。

【図15】 実施の形態4による機能ブロック図である。

【図16】 実施の形態5による機能ブロック図である。

【図17】 図16のブロックの処理手順を示すフローチャートである。

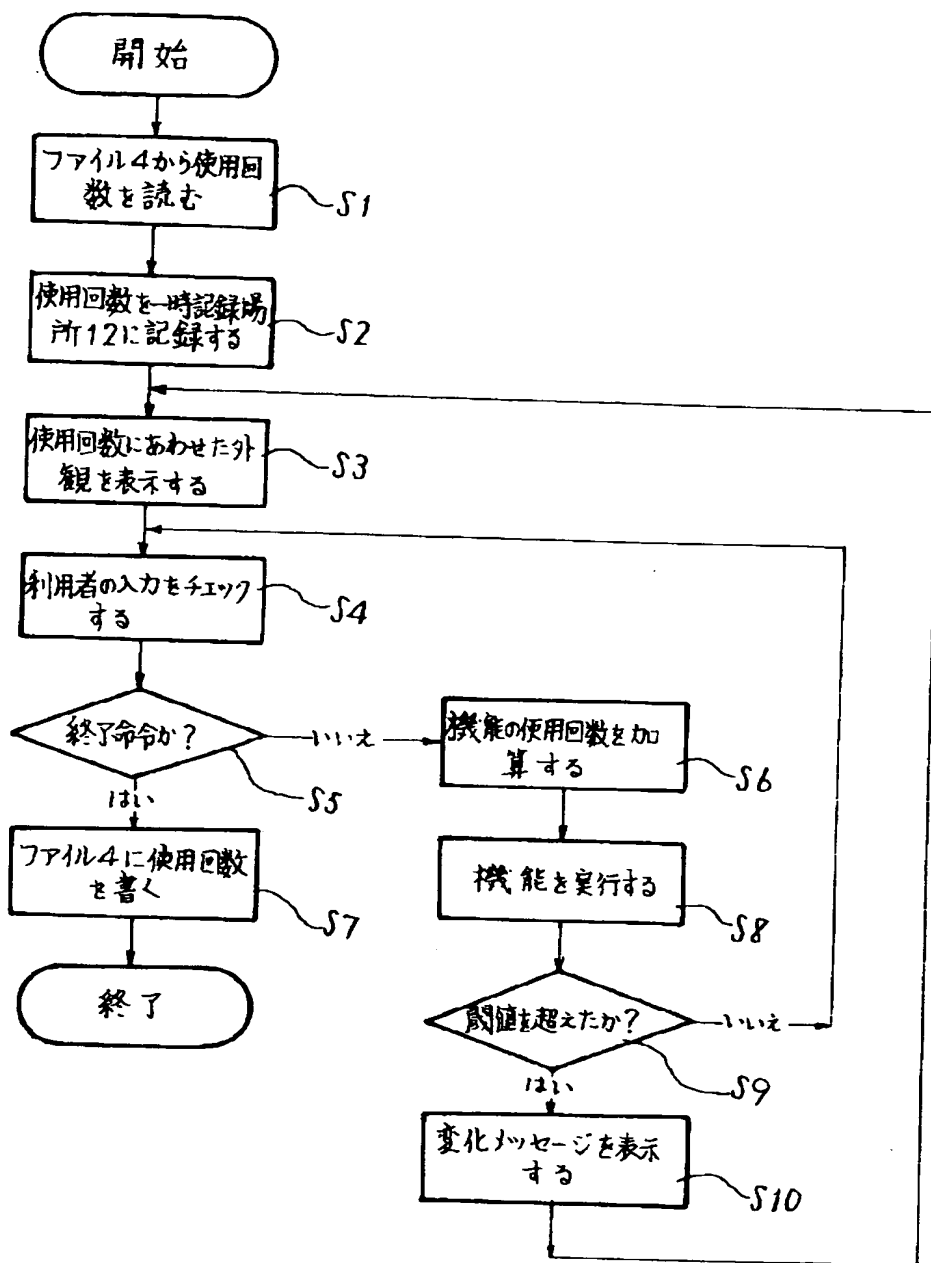
【図18】 ソフトウェアの操作画面を変化させる従来のシステム構成図である。

【図19】 図18のシステムに於けるボタン表示の説明図である。

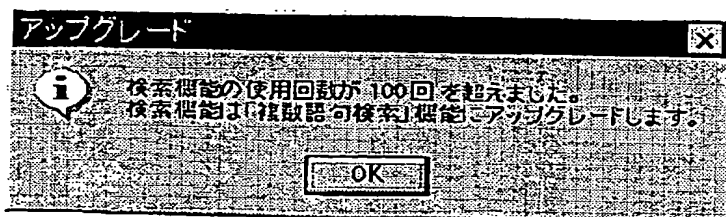
【図20】 図18のシステムの動作フローチャートである。

【図2】

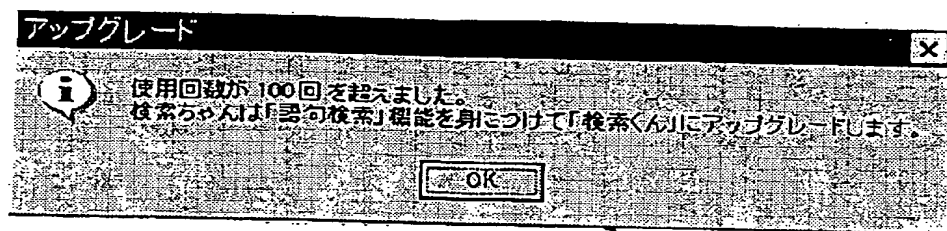
【図19】



【図8】



【図3】

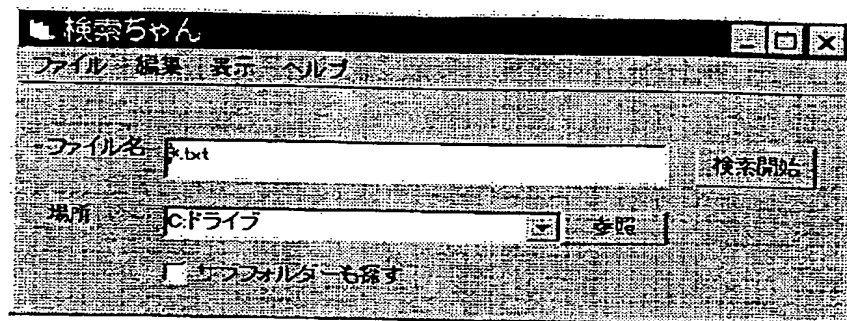


50

【図22】

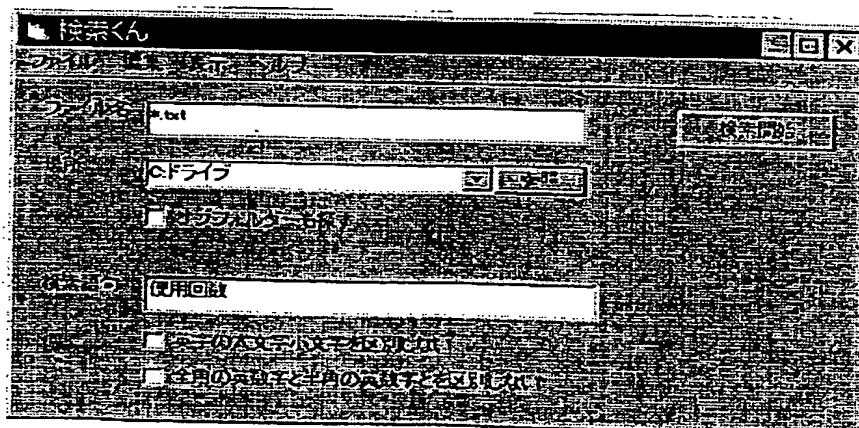
ボタン #6 アクセス回数: 12	...
ボタン #10 アクセス回数: 8	
ボタン #3 アクセス回数: 7	
...	

【図4】



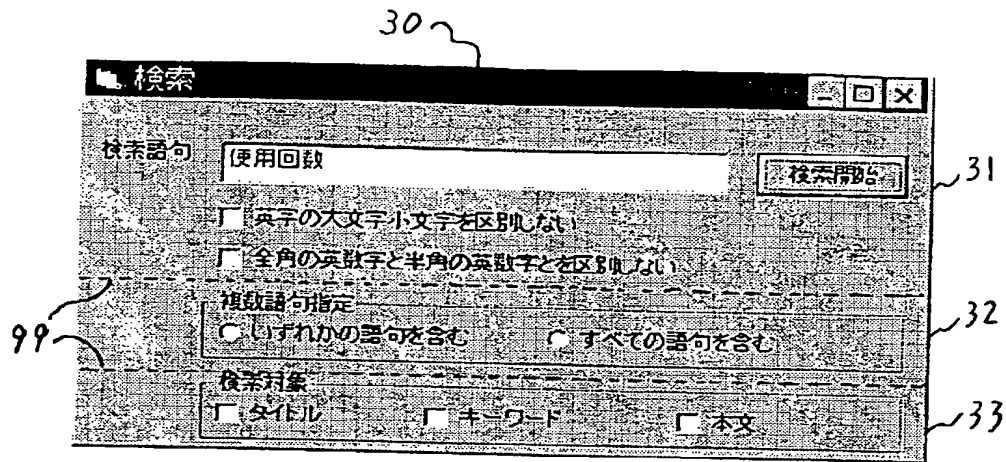
51

【図5】

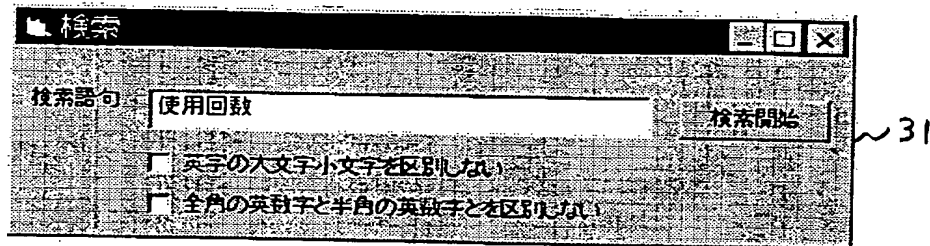


52

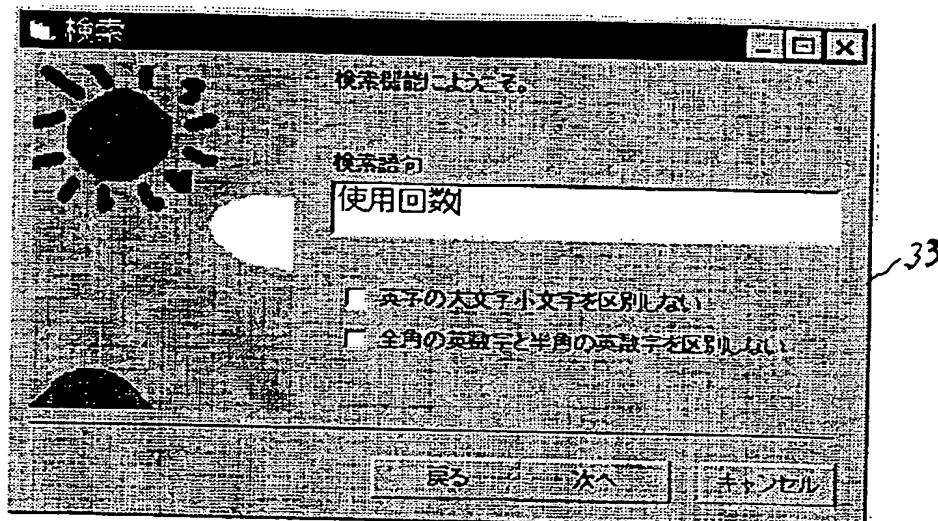
【図6】



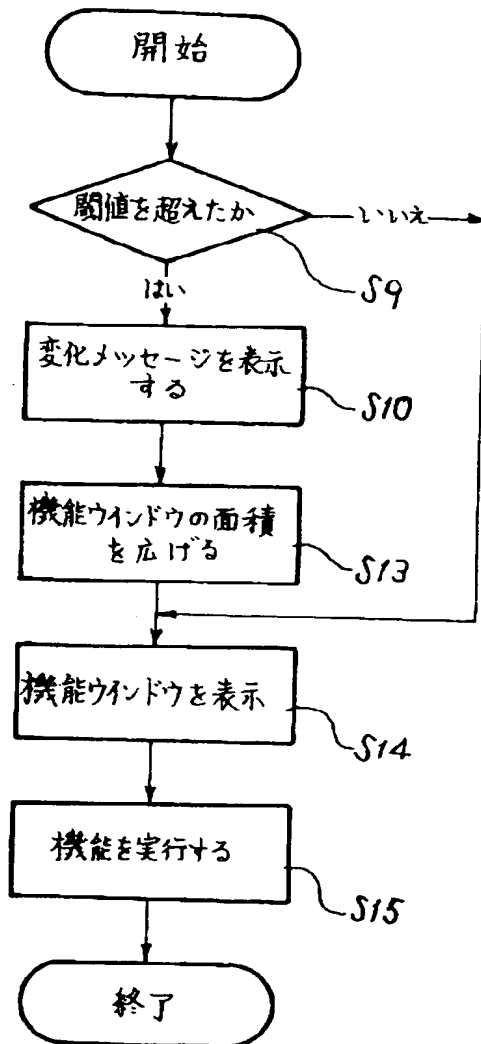
【図7】



【図10】



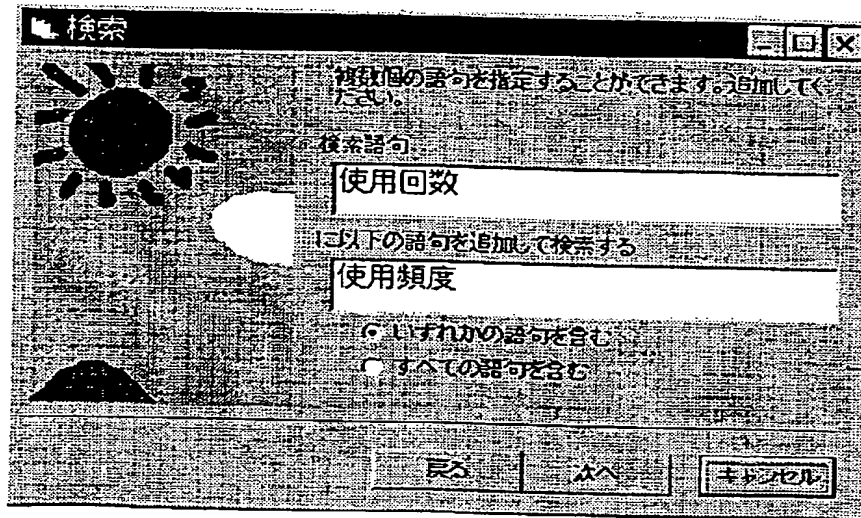
【図9】



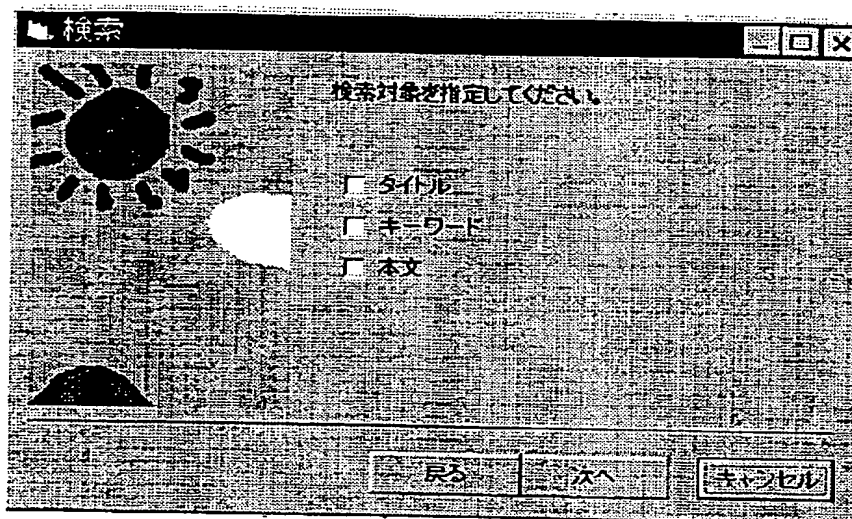
【図21】

ボタン #	ボタン # 0
0		アクセス回数
ボタン #		ボタン # 1
1		アクセス回数
ボタン #		ボタン # 2
2		アクセス回数
⋮		⋮

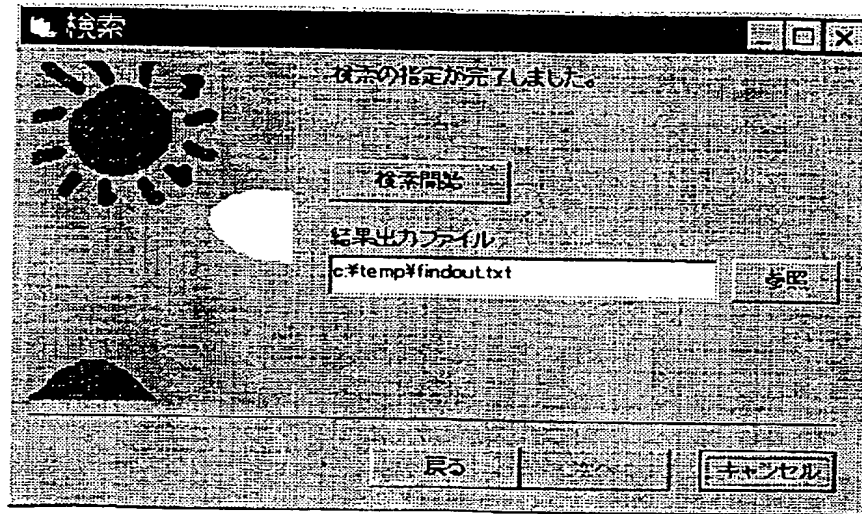
【図11】



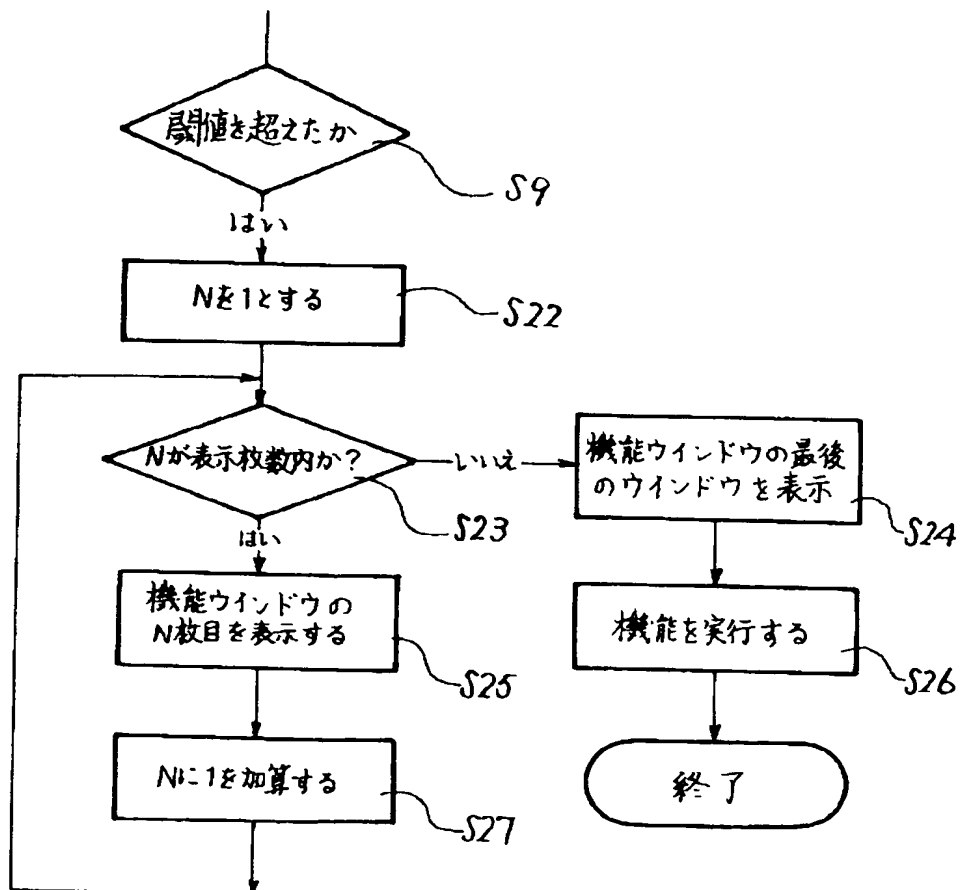
【図12】



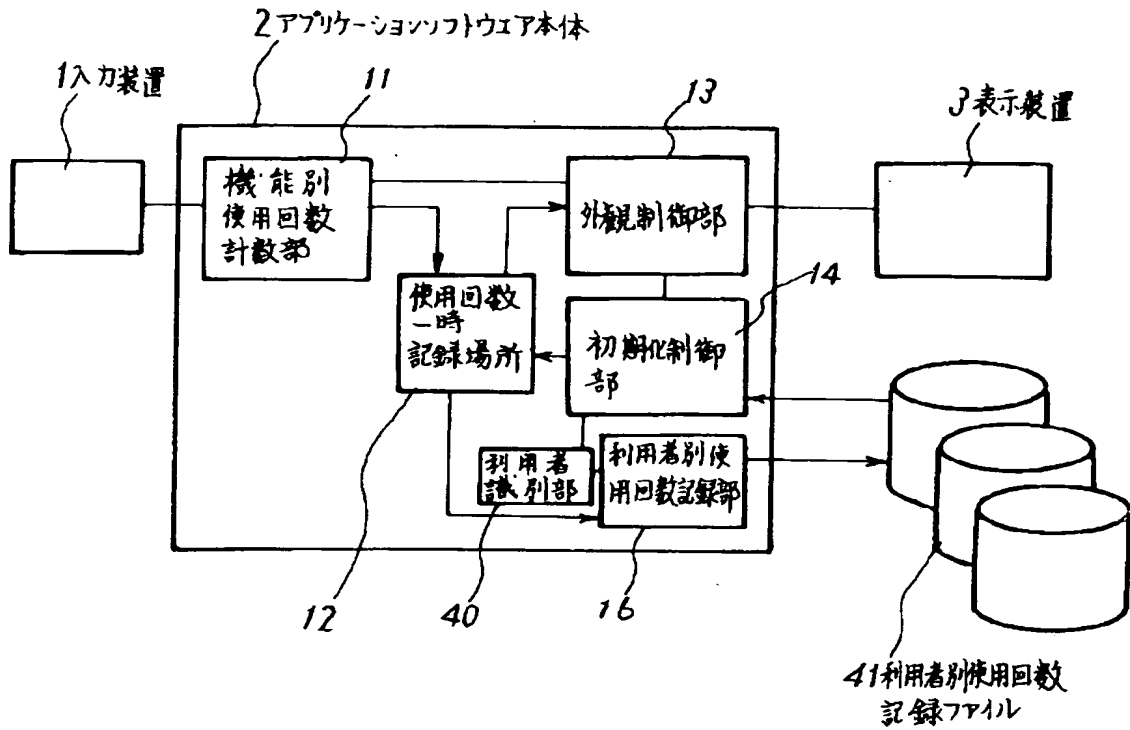
【図13】



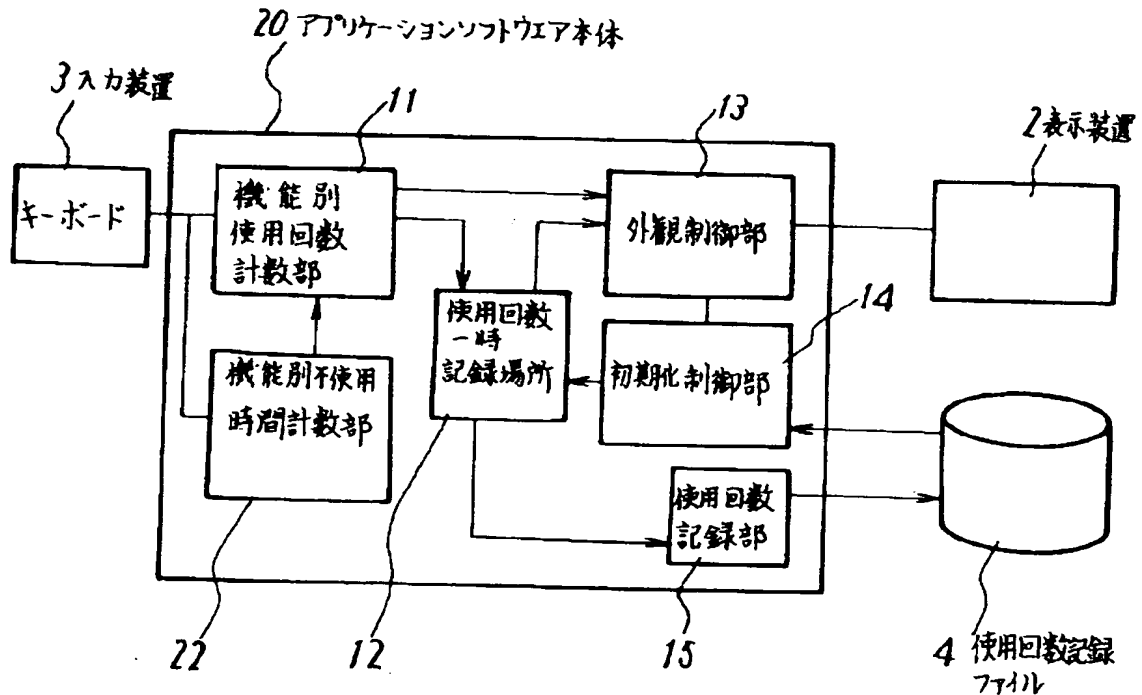
【図14】



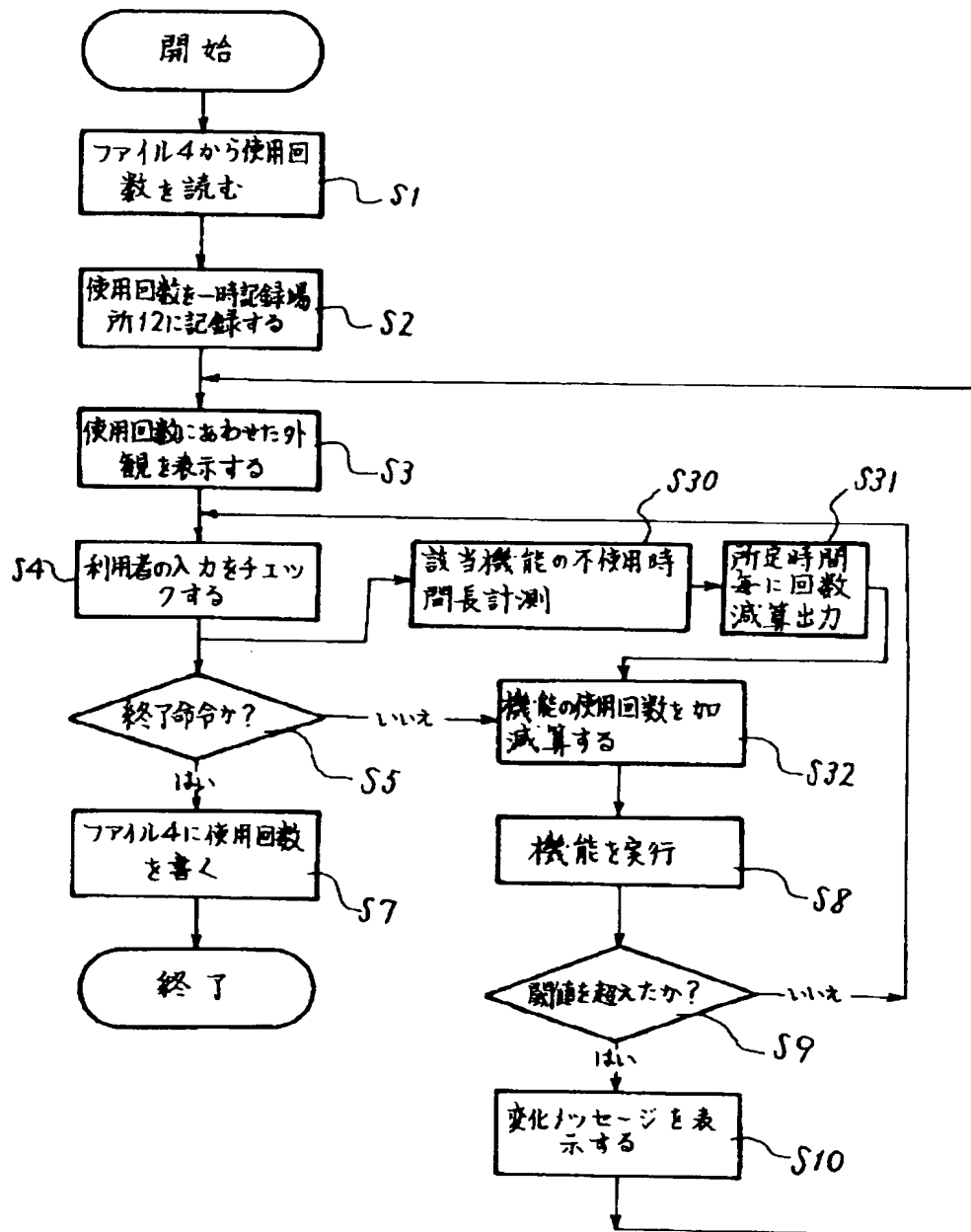
【図15】



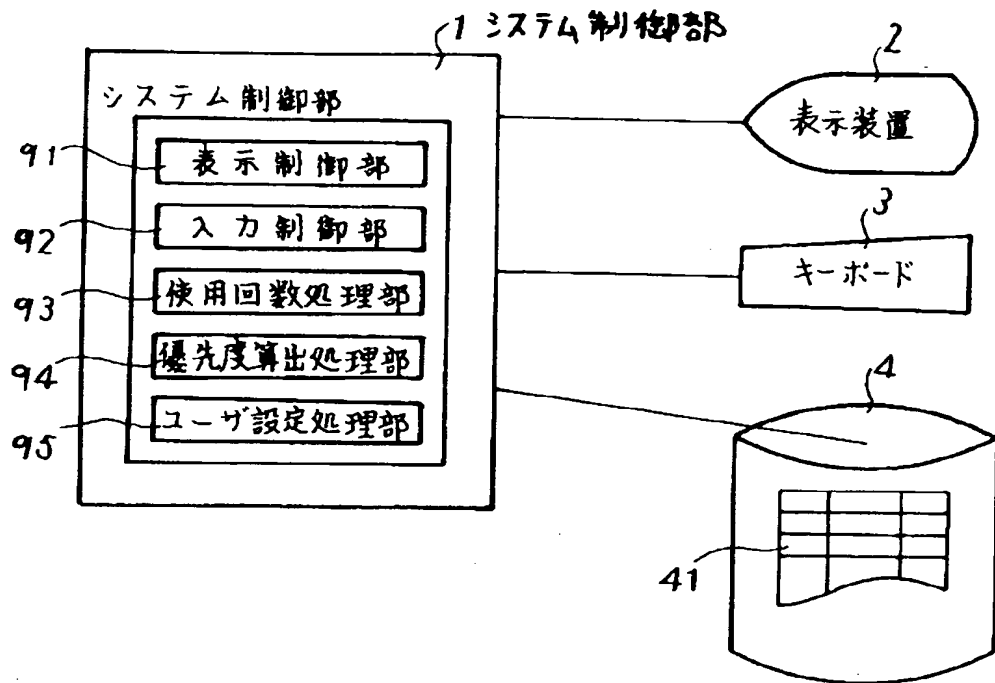
【図16】



【図17】



【図18】



【図20】

